Livingstone Academies: Problem-Solving and Computational Thinking Rubric

Introduction

- This Problem-Solving and Computational Thinking Rubric has been developed through classroom observations, seeking advice from expert and experienced teachers, and researching the 21st Century Skills and computational thinking skills domain. This rubric is a first draft effort by a group of educators on articulating learning behaviours and dispositions of learners in the classroom.

- The learning behaviours and dispositions have not been grouped by age group or phase of education because there is insufficient academic research into this field. This document should be considered in context, that is, both age and ability appropriate, and should be applied to school curriculum and subject being taught. Teachers are encouraged to adapt the rubric based on their classroom experiences and share it with the wider community.

- The Problem-Solving and Computational Thinking Rubric is focused on learners and not the concepts underpinning problem solving, critical thinking, computational thinking or other overlapping thinking skills from other subject disciplines. Instead, these concepts are implicit in the statements. The learning behaviours in the ‘realised’ column may be aspirational for learners in primary (ages 5 – 11) and early secondary (ages 11-13) education. Teachers should apply professional judgement in how much to share and how to explain new or complex vocabulary to some learners.

- This Problem-solving and Computational Thinking Rubric can be a valuable tool for teachers when planning and delivering lessons ensuring learners have opportunities to develop positive attitudes to, for example, complexity or ambiguity. This is because the response of learners to learning opportunities (behaviourism) is dependent upon the activity, and both the way in which the activities are presented to learners by the teacher (constructivism and constructionism) and how learner processes the newly acquired knowledge (cognitivism). Therefore, there is a relationship between the ‘approaches’ by teachers and the behaviours exhibited by learners’ and as such ‘approaches’ will feature in the rubric to provide some context.

- Teachers might also find the Problem-Solving and Computational Thinking Rubric useful in avoiding the use of passive verbs to describe learning outcomes (e.g. be aware of, be familiar with, know, understand, etc.) because the associated behaviours are often internal or not public, that is, they are not observable and therefore we can never really understand how the learner processes the newly acquired knowledge.
This rubric should be regarded as a tool to aid (self, peer and teacher) assessment of learners Problem-Solving in any subject. Problem solving in any subject like any skill requires repeated exposure to a range of real world problems and how they are studied, solved and represented. Therefore, this should be used periodically in conjunction with a learner portfolio of work to make an overall judgement on learners Problem-solving and Computational Thinking capabilities.

Teachers may find that the rubric can be used as an aid during lesson observations alongside an updated lesson observation form (see CAS QuickStart Computing reference below) containing computational thinking prompts because it can help mentors/coaches to consider if particular learning behaviours are or are not present in learners due to the design and delivery of teaching materials or learner attitudes and motivations.

<table>
<thead>
<tr>
<th>Learning Attributes*</th>
<th>Hibernating</th>
<th>Passive</th>
<th>Active</th>
<th>Realised</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>“Confidence in understanding problems and dealing with complexity”</strong></td>
<td>Makes no effort to attempt to grasp any aspects of the problem.</td>
<td>Gathers some of the necessary information to be able to understand the problem.</td>
<td>Gathers all necessary information to understand the problem.</td>
<td>Confidently and systematically breaks down problems and filters out unnecessary information and is able to explain the processes involved.</td>
</tr>
<tr>
<td></td>
<td>Avoids problems/challenges that have more than one step or part to solving them.</td>
<td>Grasps some but not all aspects of the problem often making educated guesses.</td>
<td>Understands all aspects of a problem.</td>
<td>Effectively organises all parts of a problem to develop coherent sub-solutions.</td>
</tr>
<tr>
<td></td>
<td>May unintentionally overly complicate problems.</td>
<td>Consistently seeks advice and reassurance.</td>
<td>Displays independence in breaking down problems and filtering out unnecessary information.</td>
<td>Sub-solutions are appropriately coupled (joined).</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Requires support on how to approach each part of the problem/challenge.</td>
<td>With support, builds solutions in parts (sub-solutions) to recompose for a final solution.</td>
<td>Reduces the complexity of the solution without affecting its behavior.</td>
</tr>
<tr>
<td></td>
<td><strong>“Persistence in working with difficult problems”</strong></td>
<td>Doesn’t engage with and avoids problems that are difficult to deal with and hard to solve.</td>
<td>Reluctantly engages with difficult problems but doesn’t persevere for long.</td>
<td>Responds positively to difficult problems and validates outcomes.</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>Displays persistence at times of difficulty.</td>
<td>Able to explain how they have overcome difficult problems.</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td>Demonstrates resilience despite setbacks.</td>
</tr>
<tr>
<td><strong>“Tolerance for ambiguity”</strong></td>
<td>• Refuses to acknowledge or struggles to accept ambiguity in both problems/solutions.</td>
<td>• Shows a willingness to acknowledge and accept some ambiguity exists in both problems/solutions.</td>
<td>• Shows an ability to tolerate ambiguity in both problems/solutions.</td>
<td>• Celebrates ambiguity and having different interpretations</td>
</tr>
<tr>
<td>-----------------------------</td>
<td>-------------------------------------------------</td>
<td>-------------------------------------------------</td>
<td>-------------------------------------------------</td>
<td>-------------------------------------------------</td>
</tr>
<tr>
<td><strong>“Deals with open ended problems”</strong></td>
<td>• Struggles to get started without a plan and clear expectations/ deliverables.</td>
<td>• Does minimum expected but no more.</td>
<td>• Investigates a limited number of problems/solutions.</td>
<td>• Applies effort to independently explore an appropriate range of problems/solutions.</td>
</tr>
<tr>
<td><strong>“Adapts existing knowledge or solutions to solve new problems”</strong></td>
<td>• Shows reluctance to or actively avoids learning from previous solutions.</td>
<td>• Struggles to identify patterns that match a problem to a previously learned solution.</td>
<td>• When directed and with reassurance, will consider adapting pre-existing solutions to solve the current problem.</td>
<td>• Tinkers with solutions to find new uses.</td>
</tr>
<tr>
<td><strong>“Iteratively develops, tests, and debug solutions”</strong></td>
<td>• Struggles to express ideas as a solution.</td>
<td>• Implements a solution using a given (completed) design.</td>
<td>• With support designs and models solutions.</td>
<td>• Independently designs, models, tests, debugs and refines solutions (using a test plan and data where appropriate).</td>
</tr>
<tr>
<td><strong>“Evaluates and weighs-up outcomes carefully”</strong></td>
<td><strong>Doesn’t consider evaluating the efficacy of a solution.</strong></td>
<td><strong>Shows an awareness of the need to evaluate solutions against criteria but requires support and advice during this process.</strong></td>
<td><strong>Independently evaluates the quality of solutions against given criteria.</strong></td>
<td><strong>Considers if a solution is ‘fit for purpose’.</strong></td>
</tr>
<tr>
<td>---</td>
<td>---</td>
<td>---</td>
<td>---</td>
<td>---</td>
</tr>
<tr>
<td><strong>“Communicates and works with others to achieve a common goal or solution”</strong></td>
<td><strong>Has a negative impact on others.</strong></td>
<td><strong>Does not have a negative impact on others.</strong></td>
<td><strong>Has a positive impact on the group.</strong></td>
<td><strong>Has a positive impact on the group.</strong></td>
</tr>
<tr>
<td></td>
<td><strong>Communicates ineffectively</strong></td>
<td><strong>Passively participates in the group, making no significant contribution.</strong></td>
<td><strong>Communicates effectively with others.</strong></td>
<td><strong>Groups working (with support) in parallel on the same problem/solution.</strong></td>
</tr>
<tr>
<td></td>
<td><strong>Makes inappropriate contributions to the group.</strong></td>
<td><strong>Uses the correct subject vocabulary.</strong></td>
<td><strong>Groups working (with support) in parallel on the same problem/solution.</strong></td>
<td><strong>Makes positive contributions to and supports others.</strong></td>
</tr>
<tr>
<td></td>
<td><strong>Uses incorrect subject vocabulary.</strong></td>
<td><strong>Has a sense of own strengths and weaknesses but evidences little strategy to deal with them.</strong></td>
<td><strong>Explains how a solution works to others.</strong></td>
<td><strong>Understands own strengths and weaknesses and solicits help from appropriate others.</strong></td>
</tr>
<tr>
<td></td>
<td><strong>Has no sense of own strengths and weaknesses or those of others.</strong></td>
<td><strong>With prompts can explain how a solution works to others.</strong></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

* Column a used inspiration for, and adapted from, the CTSA Operational definition of Computational Thinking for K12 Education
Bibliography

BBC Bitesized (2016), Computational Thinking
http://www.bbc.co.uk/education/topics/z7tp34j

Barr, V. & Stephenson, C. (2011), Bringing Computational Thinking to K12

Bhattacharya, N., (2016), Tinkering (an introduction)
https://vimeo.com/182482442

Brennan, K., & Resnick, M. (2012), New frameworks for studying and assessing the development of computational thinking

Computing At School (2014), CAS Barefoot Computational Thinking
http://barefootcas.org.uk/barefoot-primary-computing-resources/concepts/computational-thinking/

https://community.computingatschool.org.uk/resources/2324

Computing At School (2015), Lesson observation form with prompts
http://www.quickstartcomputing.org/secondary/section4.html

CSTA (2011), Operational definition of Computational Thinking for K12 Education
https://csta.acm.org/Curriculum/sub/CurrFiles/CompThinkingFlyer.pdf

CSTA Voice (2016), Irene Lee: Reclaiming the Roots of CT, Page 4
https://csta.acm.org/Communications/sub/CSTAVoice_Files/csta_voice_03_2016.pdf

Department for Education (2014), Computing Programmes of Study

Created by: Mark Dorling and Thomas Stephens
Tested and edited by: Teachers form Aspirations Academies Trust, form the Digital Schoolhouse satellite schools, Phil Bagge, Andrew Csizmadia, Pete Marshman, Thomas Ng, Shahnella Saeed and Dr. Chris Stephenson

Feedback and review from the wider Initial Teacher Training and Computing Education community.

Copyright 2016 Livingstone Academies part of the Aspirations Academies Trust
https://community.computingatschool.org.uk/resources/1692

Fuller et. Al. (2007), *Developing a computer science-specific learning taxonomy*  
http://dl.acm.org/citation.cfm?id=1345438

Google (2016), *Computational Thinking for Educators*  
https://computationalthinkingcourse.withgoogle.com/unit

Google (2016), *Computational Thinking from a Dispositions Perspective*  
http://googleforeducation.blogspot.co.uk/2016/09/Computational-Thinking.html?m=1

Google (2016), *Exploring Computational Thinking*  

Potter. M, and Kustra (2012), *A primer on learning outcomes and the SOLO Taxonomy*  

*A focus on writing learning outcomes*  
https://afocusonlearningoutcomes.wordpress.com/resources/extras/

STEM Learning, *Computational Thinking*  
https://www.stem.org.uk/elibrary/resource/35192